



googless

Copyright Notice

Slides and Notes Licensed as:

- **AU Creative Commons 2.5**
 - Attribution-Non Commercial-No



Derivative Works

Code Licensed as:

- **Apache License, Version 2.0**

Updates to Slides

Incorporates all previous slides from:

- OWASP USA Conference 2008
- ToorCon X (USA)
- SecTor 2K8 (Canada)
- RUXCON 2K8 (Australia)
- OWASP Australian Conference 2009
- OWASP European Conference 2009
- 5th CONFidence 2009 (Poland)
- OWASP London Chapter Meeting May 2009

Lastest Updated **25 June 2009**

Latest (SFW) Slides

Published on  slideshare

http://www.snipurl.com/cmlh_slideshare

Published as Separate PPT Presentations

- "SyScan09SG" Tag

Slides will be updated post SyScan'09 SG

References and Further Info

Refer to the **Notes** Page of each Slide

Some slides are hidden due to time limit

Christian Heinrich aka “cmlh”

“End User” Experience since 1996:

- Current **State .nsw.gov.au**
 - Past Critical Infrastructure (Utilities – Electricity)
- Past **Federal .gov.au**
 - **DSD** Certified Gateway Service Provider
 - ASIO Web Hosting
 - Government Endorsed Business (GEB)

Linked  profile <http://www.linkedin.com/in/ChristianHeinrich>

Christian Heinrich aka “cmlh”

“End User” Experience since 1996:

- Security Thought Leader within AU **Media**:
 - Former CSO of FOXTEL
 - Former CSO of News Limited (AU part of News Corp)

Linked  profile <http://www.linkedin.com/in/ChristianHeinrich>

Agenda

1. OWASP Google Hacking Project

1.1 "Search Engine Recon/Discovery"

1.2. "Download Indexed Cache"

2. "TCP Input Text"

3. OWASP Google Hacking Project

3.1 "Spiders/Robots/Crawlers"

3.2 "Continuous Improvement"



OWASP "Google Hacking" Project

Christian Heinrich

christian.heinrich@owasp.org

OWASP Project Lead

SyScan'09 Singapore

Copyright © The OWASP Foundation

Permission is granted to copy, distribute and/or modify this document
under the terms of the OWASP License.

The OWASP Foundation

<http://www.owasp.org>

Contributions to OWASP

OWASP Testing Guide v3

- 4.2.1 “Spiders/Robots/Crawlers”
- 4.2.2 “Search Engine Reconnaissance”

OWASP “Google Hacking” Project

- “Download Indexed Cache” PoC

Presented at OWASP AU, EU and USA Conferences





OWASP Testing Guide v3

4.2.2 "Search Engine Discovery"

Christian Heinrich

christian.heinrich@owasp.org

OWASP Project Lead

SyScan'09 Singapore

Copyright © The OWASP Foundation

Permission is granted to copy, distribute and/or modify this document
under the terms of the OWASP License.

The OWASP Foundation

<http://www.owasp.org>

How Google Indexes My Site:?

- Crawled by “Googlebot”:
 - ▶ Sitemap provided by Webmaster
- Indexed and (maybe) Retained in Google Cache
- Repeat

Residual Risk

Your **site**: Published by Google Cache

Examples:

- Company Annual Reports prior to Public Release
- Firewall Rules from ANY Source
 - ▶ UAT during development of Web Application
 - ▶ Web Administration Portal of CMS
- Third Party Reproduction
 - ▶ "*Who is ecopeland ...?*"

Residual Risk

Advantages to an Attacker during “Recon” Phase:

1. “Sort of” Anonymous Attack Surface Mapping
 - ▶ Images, etc are downloaded by Web Browser
2. Page Rank Orders “Less Public” Results Last

Google Hacking Database (GHDB) Approach

<http://johnny.ihackstuff.com/ghdb.php>

Examples:

■ Microsoft Remote Desktop Web Connection

▶ `intitle:Remote.Desktop.Web.Connection inurl: tsweb`

■ Virtual Network Computer (VNC)

▶ `"VNC Desktop" inurl:5800`

■ Outlook Web Access

▶ `inurl:"exchange/logon.asp"`

▶ `intitle:"Microsoft Outlook Web Access - Logon"`

Issues with the GHDB Approach



Issues with the GHDB Approach

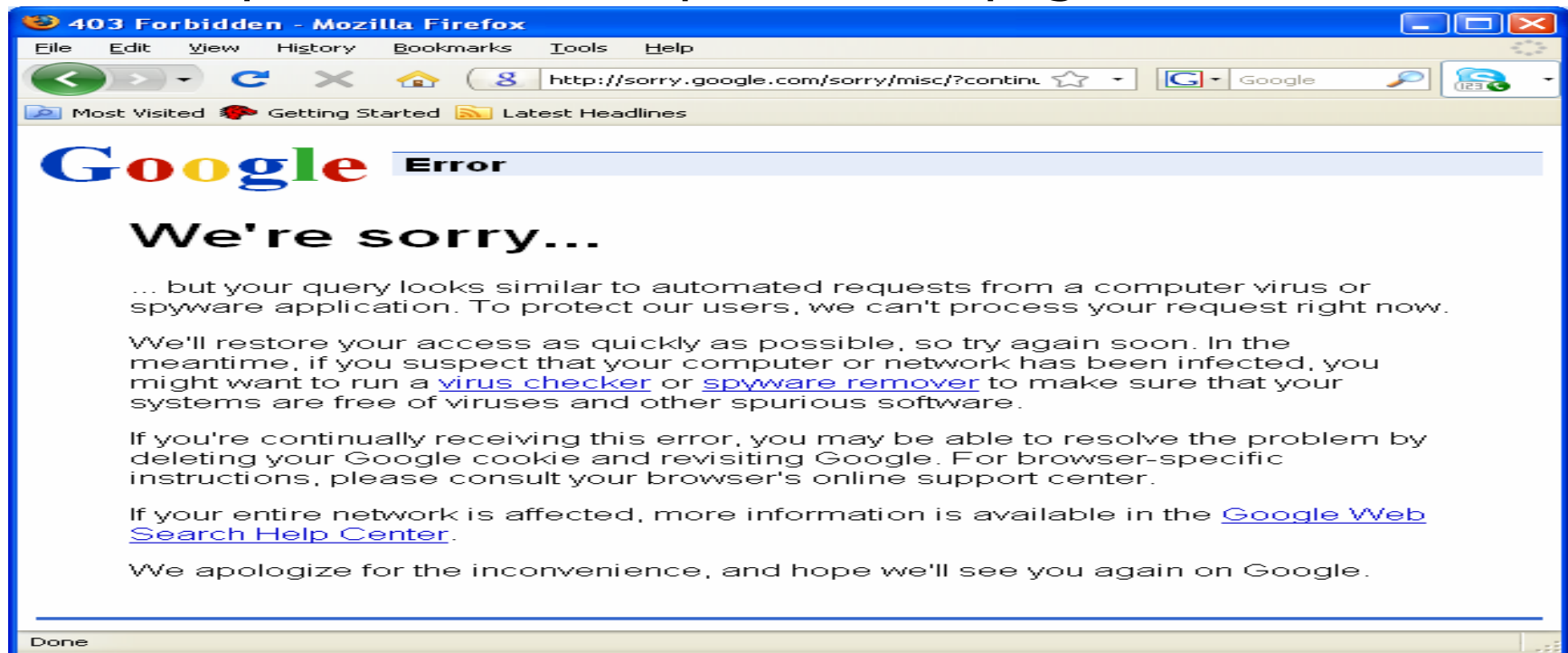


Issues with GHDB Approach

Intended for Search Engine Worms only.

■ “We’re Sorry”

▶ Implemented to Stop Worm Propagation



Issues with GHDB Approach

Misses Majority of Vulnerability and Info Leakage

- Limited by number of signatures available

Not targeted to site:

- Signatures are for all sites: without modification

Many False Positives

Resolved in OWASP Testing Guide v3

OWASP Testing Guide v3 Methodology

Targets **all** Vulns and Information Leakage

- No False Positives

- ▶ Only Human Error 😊

Based on Google Advanced Search Operators:

- **site:**

- **cache:**

Google Advanced Search `site:` Operator

Targeted to a Specific site:

`site:www.google.com`

1. `www.google.com` only



Google Advanced Search `site:` Operator

Targeted at all hostnames and sub domains:

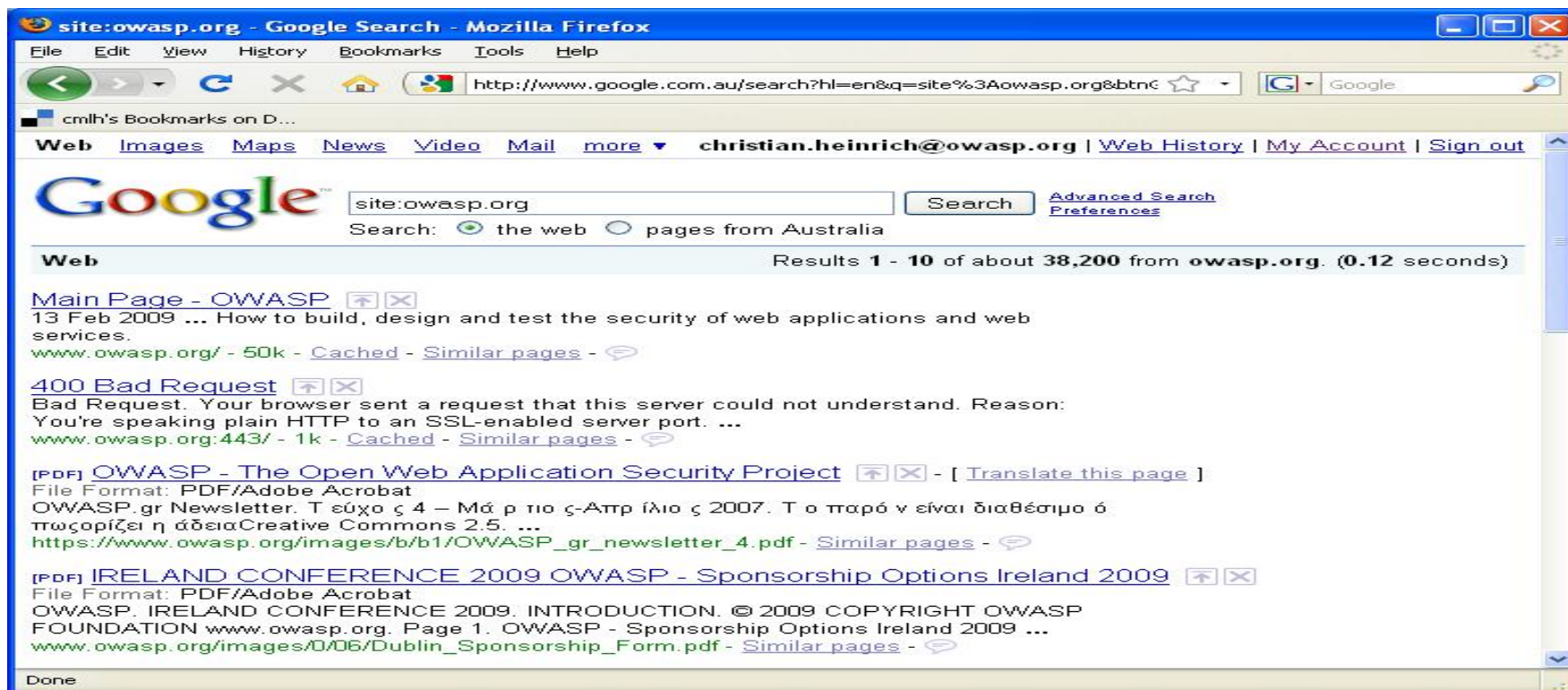
`site:google.com`

1. `www.google.com`
2. `video.google.com`
3. `code.google.com`



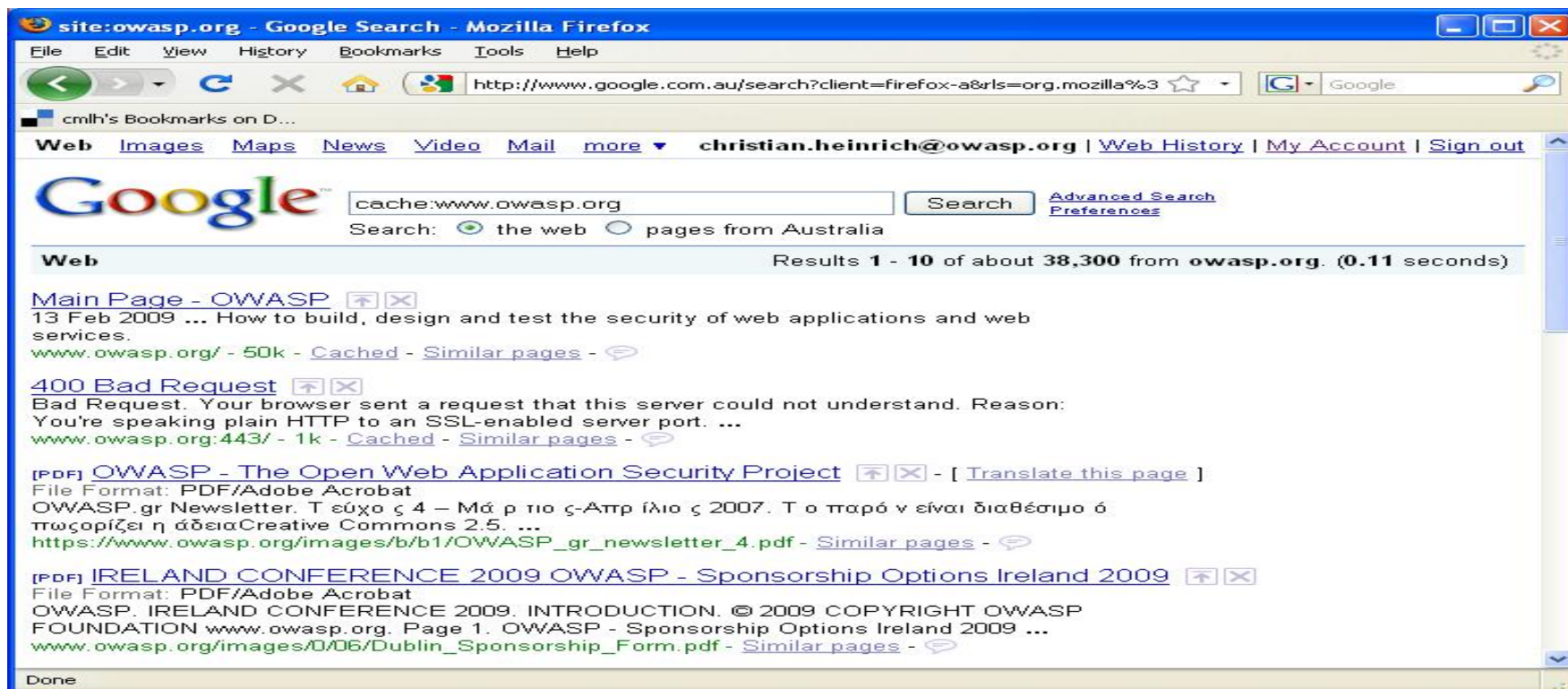
Google Advanced Search cache: Operator

Click "*Cached*"



Google Advanced Search cache: Operator

cache:www.owasp.org



Google Advanced Search cache: Operator





Download Indexed Cache

Christian Heinrich

christian.heinrich@owasp.org

OWASP Project Lead

SyScan'09 Singapore

Copyright © The OWASP Foundation

Permission is granted to copy, distribute and/or modify this document
under the terms of the OWASP License.

The OWASP Foundation

<http://www.owasp.org>

Download Indexed Cache

Part of the OWASP "Google Hacking" Project

Repository <http://code.google.com/p/dic>

Supports OWASP Testing Guide v3

- 4.2.2 "Search Engine Reconnaissance"

Implements the Google SOAP Search API

- Search and Cache related SOAP Messages

Command Line Arguments

Google SOAP Search API related:

- **-key** API Key
demo is embedded API Key
- **-query** Google Search Query
- **-start** Starting Google Search Result
(Zero Based Index i.e. 1=0)

Results 1 to 10

```
cmlh$ /usr/bin/perl dic.pl -key "demo" -query "site:owasp.org" -start 1

"Download Indexed Cache" Proof of Concept (PoC) 0.1 (Released at RUXCON 2K8)

Copyright 2009 Christian Heinrich
Licensed under the Apache License, Version 2.0

Creating ./siteowasp.org

1. Downloading https://www.owasp.org/ from Google Cache [46k] as 1.html
2. Downloading http://www.owasp.org/ from Google Cache [46k] as 2.html

[SNIP]

8. Downloading http://www.owasp.org/index.php/Session_Management from
   Google Cache [88k] as 8.html
9. Downloading http://www.owasp.org/index.php/Testing_for_file_extensions
   handling from Google Cache [24k] as 9.html
10. Downloading http://www.owasp.org/index.php/OWASP_SoC_2008_AS DR_Reviewers
    from Google Cache [20k] as 10.html
```

Results 11 to ...

```
cmlh$ /usr/bin/perl dic.pl -key demo -query "site:owasp.org" -start 11
```

```
"Download Indexed Cache" Proof of Concept (PoC) 0.1
```

```
Copyright 2008 Christian Heinrich
```

```
Licensed under the Apache License, Version 2.0
```

```
Appending ./siteowasp.org
```

- 11. Downloading https://www.owasp.org/index.php/System_Information_Leak from Google Cache [26k] as 11.html
- 12. Downloading http://www.owasp.org/index.php/Buffer_overflows from Google Cache [34k] as 12.html

```
[SNIP]
```

- 18. Downloading http://www.owasp.org/index.php/Testing_Guide_Introduction from Google Cache [111k] as 18.html
- 19. Downloading http://www.owasp.org/index.php/OWASP_Java_Project from Google Cache [28k] as 19.html
- 20. Downloading https://www.owasp.org/index.php/Insecure_Temporary_File from Google Cache [26k] as 20.html



Google Search Results - 1 to 1000

```
#!/usr/bin/perl -w
for (my $result=0; $result < 990; $result = $result + 10) {
    system ("./dic.pl -key \"[key]\" -query \"[query]\" -start $result\n");
}
```

Exploiting Page Rank

Page Rank Orders “Less Public” Results Last

Descending `$start` of `doGoogleSearch`:

- e.g. `-start:990`, `-start:980`, etc
- Remember `$start - 1` i.e. 0

Google Search Results - 1000 to 1

```
#!/usr/bin/perl -w
for (my $result=990; $result >= 1; $result = $result - 10) {
    system ("../dic.pl -key \"[key]\" -query \"[query]\" -start $result\n");
}
```

Generated Output

```
cmlh$ /usr/bin/perl dic.pl -key "demo" -query "site:owasp.org" -start 1

"Download Indexed Cache" Proof of Concept (PoC) 0.1 (Released at RUXCON 2K8)

Copyright 2009 Christian Heinrich
Licensed under the Apache License, Version 2.0

Creating ./siteowasp.org

1. Downloading https://www.owasp.org/ from Google Cache [46k] as 1.html
2. Downloading http://www.owasp.org/ from Google Cache [46k] as 2.html

[SNIP]

8. Downloading http://www.owasp.org/index.php/Session_Management from
   Google Cache [88k] as 8.html
9. Downloading http://www.owasp.org/index.php/Testing_for_file_extensions
   handling from Google Cache [24k] as 9.html
10. Downloading http://www.owasp.org/index.php/OWASP_SoC_2008_ASDR_Reviewers
    from Google Cache [20k] as 10.html
```



Generated Output

Directory

- Name Stripped of “.” from Google Operator
- /dic sub-directory

Files in Directory:

- *x.html*
 - ▶ *x* is Search Result Number
- [*SearchQuery*].csv
 - ▶ *SearchResultNumber, URL*

1.html Example

```
cmlh$ cd siteowasp.org/dic/  
cmlh$ head -n 25 1.html
```

```
<meta http-equiv="Content-Type" content="text/html;  
charset=UTF-8"><base href="https://www.owasp.org/index.php  
/Main_Page"><div style="margin:-1px - 1px 0;padding:  
0;border:1px solid #999;background:#fff"><div style=  
"margin:12px;p adding:8px;border:1px solid  
#999;background:#ddd;font:13px arial,sans-  
serif;color:#000;font-weight:normal;text-align:left">This  
is Google's cache of <a href="https://www.owasp.org/"  
style="text decoration:underline;color:#00c">https://  
www.owasp.org/</a>. It is a snapshot of the page as it  
appeared on 17 Feb 2009 17:00:03 [snip]
```


[*SearchQuery*].csv Example

```
cmlh$ cat siteowasp.org.csv
1,http://www.owasp.org/
2,http://www.owasp.org/download/
3,http://www.owasp.org:443/
4,https://www.owasp.org/images/b/b1/OWASP_gr_newsle [snip]
5,http://www.owasp.org/images/0/06/Dublin_Sponsorsh [snip]
6,https://www.owasp.org/images/2/21/OWASP_gr_newsle [snip]
7,http://www.owasp.org/index.php/Cincinnati
8,http://www.owasp.org/index.php/Testing_for_file_e [snip]
9,http://www.owasp.org/index.php/OWASP_SoC_2008_ASD [snip]
10,http://www.owasp.org/index.php/OWASP_Taiwan_Tran [snip]
```

DataDumper.txt Example

```
$VAR1 = bless( {  
  'searchTime' => '0.136083'  
  'endIndex' => '10',  
  'searchComments' => '',  
  'documentFiltering' => 0,  
  'searchTips' => '',  
  'estimatedTotalResultsCount' => '41100',  
  'searchQuery' => 'site:owasp.org',  
  'startIndex' => '1',  
  'resultElements' => [  
    bless( {  
      [SNIP]
```

Google SOAP Search API in Perl

`doGoogleSearch`

- `$key`
- `$q`
- `$start -1` subtracted for Zero Index

`doGoogleSearchResponse`

- `URL`
- `cachedSize`

Google SOAP Search API in Perl

`doGetCachedPage`

- `$key`
- `$URL`

`doGetCachedPageResponse`

- ... *`xsi:type="ns2:base64">`*

Google SOAP Search API Limitations

Search Query limited to:

- **10 Words**

- **2048 Bytes**

1K Search Queries Per Day

Limited to Search Results within **0...999**

10K Possible Results from 10 Different Queries

"10K Possible Results from 10 Different Queries"

Specific each FQDN over 10 site: -queries

For example:

1. ... *-query "site:www.google.com"* ...
2. ... *-query "site:video.google.com"* ...
3. ... 9. [snip]
10. ... *-query "code.google.com"* ...

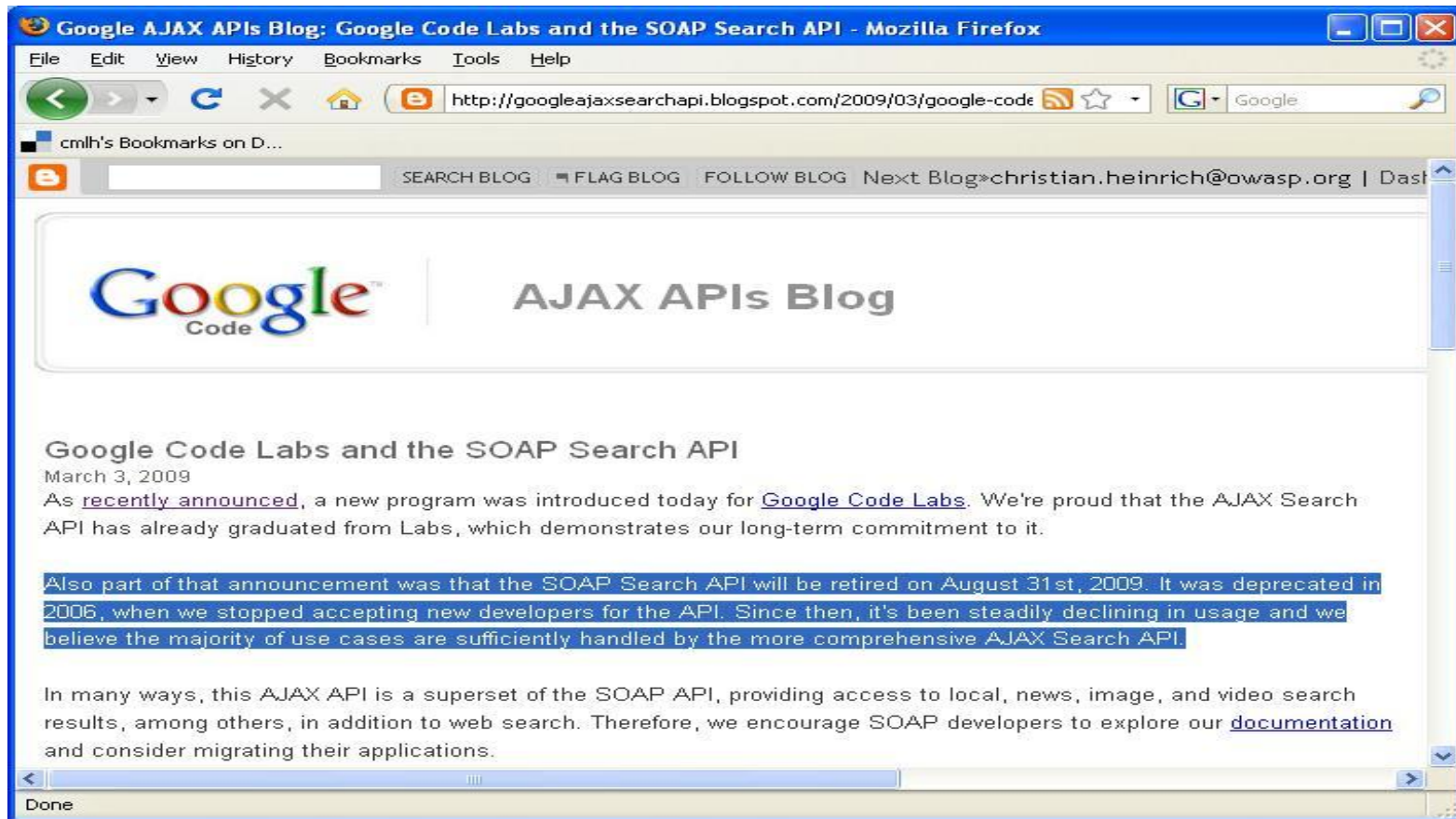
Google SOAP Search API Limitations

Issuing of API Keys Discontinued 5 Dec 2006



Google SOAP Search API Limitations

Will be Deprecated on 31 August 2009



dic Roadmap

PoC v0.1

- Previewed at OWASP USA, ToorCon and SecTor (CA)
- Released at RUXCON 2K8 in Sydney, AU, Nov 2008

PoC v0.2

- Moving repository to `code.google.com/p/dic`
- Records the Timestamp from Google Cache
- Previewed at OWASP AU and EU 2009 (in dev)

dic Roadmap

PoC v0.3

- Specify Range of Google Search Results to 1000
 - ▶ Code Sync with "TCP Input Text"
 - ▶ Consider Net::Google CPAN Perl Module



dic Roadmap

PoC v0.4

- Maintenance Release

- Released approx 31 August 2009

 - ▶ Once Google deprecates SOAP Search API

- May be relocated to separate OWASP Project

 - ▶ For Historical Archive

 - ▶ "Don't Drop the SOAP" OWASP Project Proposal

 - ▶ Refocus OWASP "Google Hacking" Project

 - Google Search Queries separate from GHDB.

Project Controversy

Linkedin® - OWASP “**Google Hacking**” Role:

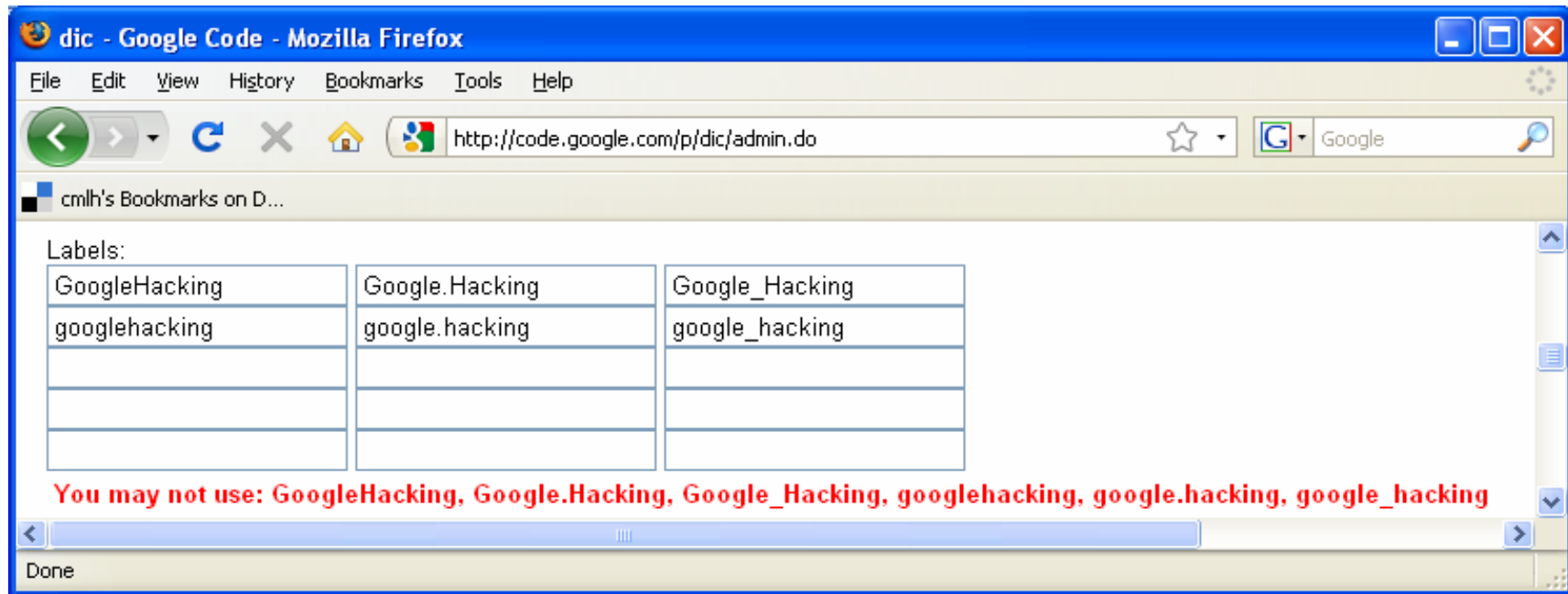
1. Someone in an Engineering Function at Google
2. Complaint Received by Tom Brennan (OWASP)

Facts:

- Not an Google or OWASP Summer of Code
- Does not violate Google’s Terms of Service
- Contacted for Sec. Role at Google Sydney AU
- Google SOAP API perl code related to `tit`
 - ▶ Separation with OWASP Project due to new scope

Project Controversy

`code.google.com` denies "*Google Hacking*" labels

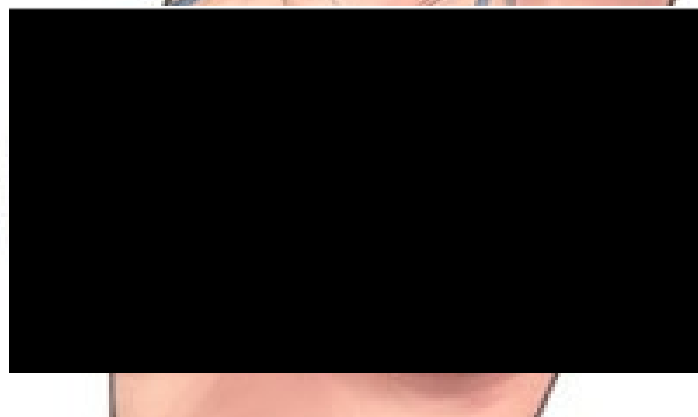


But permits project names of "*Google Hacking*"
`http://code.google.com/p/googlehacking`



TCP Input Text

Christian Heinrich aka "cmlh"



TCP Input Text

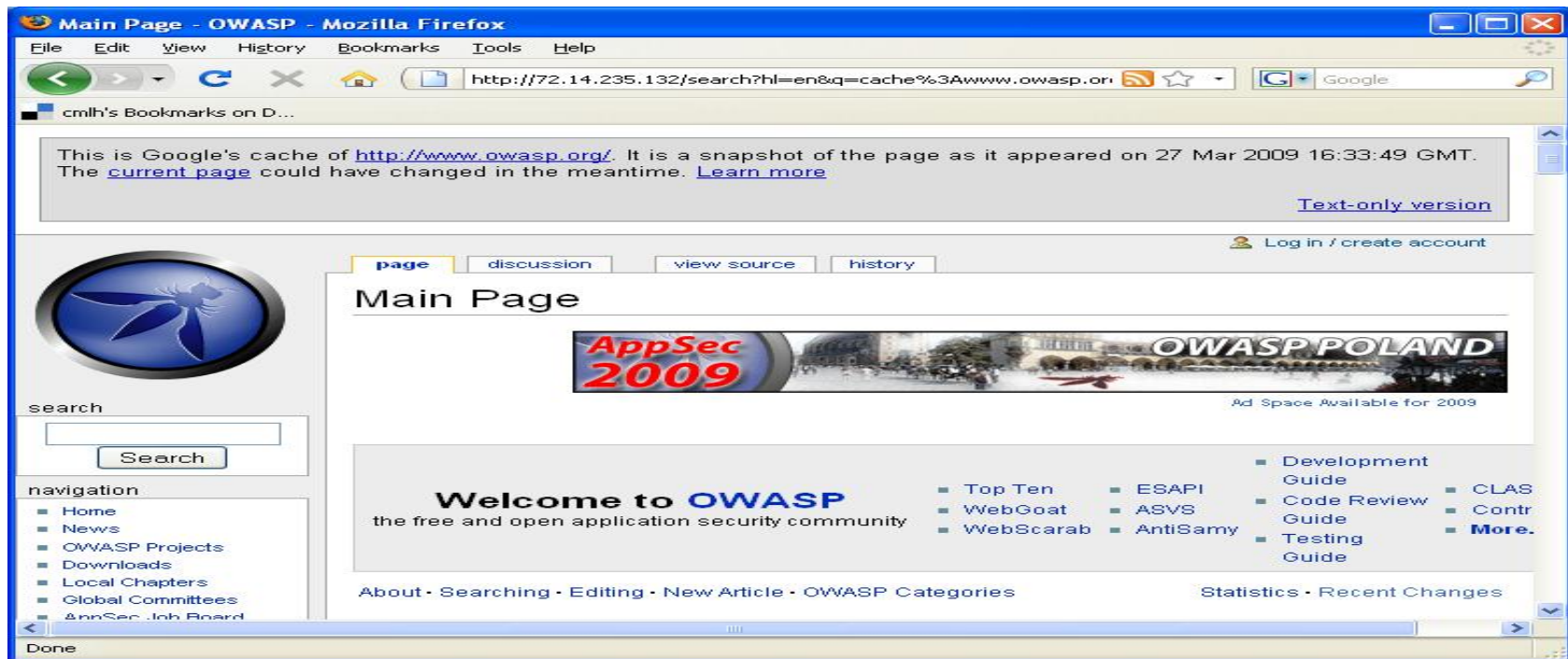
Used in the "Recon" Phase of Penetration Test

Enumerates FQDN and TCP Ports

Implements the Google SOAP Search API

cache:

May be out of date since last crawl by "Googlebot"



Hence, positive assurance by nmap, nc, etc

Output Plug-In Architecture

TCP Port and FQDN

- **nmap**
- netcat aka **nc**
- **.CSV File**

Output Files written to `./[query]/tit`

Output Plug-In Architecture

FQDN .CSV File

- Scan for TCP Ports not within Search Results.
- Forward Lookup with `dig` and/or `nslookup`

Output Files written to `./[query]/tit`

Output Plug-In Architecture

To remove duplicate entries

```
cmlh$ sort file | uniq > nodups_file
```

Output Files written to `./[query]/tit`

Regular Expression

```
=~m| (\w+) :// ([^/:]+) (: \d+)? / (.*) |;
```

```
my $Protocol = $1;
```

```
my $Domain_Name = $2;
```

```
my $URI = ("/" . $4); // discarded
```

```
if ($3 =~ /\: (\d+) /) {$TCP_Port = $1}
```

```
else {$TCP_Port = 80}
```

tit_FQDN.csv Example

```
glcfapp.umiacs.umd.edu  
www.speedguide.net  
inside.c-spanarchives.org  
www.wsu.edu  
torrents.freebsd.org  
sammelpunkt.philo.at  
arc.cs.odu.edu  
www.ripn.net  
phy043.tours.inra.fr  
202.188.95.52
```

tit_FQDN_TCP.csv Example

```
glcfapp.umiacs.umd.edu,8080  
www.speedguide.net,8080  
inside.c-spanarchives.org,8080  
www.wsu.edu,8080  
torrents.freebsd.org,8080  
sammelpunkt.philo.at,8080  
arc.cs.odu.edu,8080  
www.ripn.net,8080  
phy043.tours.inra.fr,8080  
202.188.95.52,8080
```

tit_nc.sh Example

```
nc -vz glcfapp.umiacs.umd.edu 8080
nc -vz www.speedguide.net 8080
nc -vz inside.c-spanarchives.org 8080
nc -vz www.wsu.edu 8080
nc -vz torrents.freebsd.org 8080
nc -vz sammelpunkt.philo.at 8080
nc -vz arc.cs.odu.edu 8080
nc -vz www.ripn.net 8080
nc -vz phy043.tours.inra.fr 8080
nc -vz 202.188.95.52 8080
```

tit_nmap.sh Example

```
nmap -PN -sT -p T:8080 glcfapp.[snip]  
nmap -PN -sT -p T:8080 www.spee[snip]  
nmap -PN -sT -p T:8080 inside.c[snip]  
nmap -PN -sT -p T:8080 www.wsu.edu  
nmap -PN -sT -p T:8080 torrents[snip]  
nmap -PN -sT -p T:8080 sammelpu[snip]  
nmap -PN -sT -p T:8080 arc.cs.odu.edu  
nmap -PN -sT -p T:8080 www.ripn.net  
nmap -PN -sT -p T:8080 phy043.t[snip]  
nmap -PN -sT -p T:8080 202.188.95.52
```


tit Roadmap

PoC v0.1 (RUXCON 2K8 in Sydney, AU, Nov 2008)

- Previewed at ToorCon(US) and SecTor (CA)

PoC v0.2 (SyScan'09 Singapore)

- Moving repository to `code.google.com/p/tit`
- Added FQDN Output Plug-In
- Previewed at OWASP AU 2009 (February) and 5th CONFidence 2009 (Poland)

Changes from v0.1 to v0.2

```
cmlh$ ./tit.pl -key "demo" -query "inurl:8080" -start 1
```

"TCP Input Text" PoC v0.2

Copyright 2008, 2009 Christian Heinrich

Licensed under the Apache License, Version 2.0

Output Plug-Ins (TCP_FQDN_CSV, FQDN_CSV, NMAP_SH, NC_SH)

1. glcfapp.umiacs.umd.edu TCP/8080 available
2. www.speedguide.net TCP/8080 available
3. www.wsu.edu TCP/8080 available
4. inside.c-spanarchives.org TCP/8080 available
5. torrents.freebsd.org TCP/8080 available
6. sammelpunkt.philo.at TCP/8080 available
7. arc.cs.odu.edu TCP/8080 available
8. www.ripn.net TCP/8080 available
9. phy043.tours.inra.fr TCP/8080 available

[SNIP]

tit Roadmap

v0.3 Alpha

- `sub output_plugin`
- Specify Range of Google Search Results to 1000
 - ▶ Code Sync with "Download Indexed Cache"
 - ▶ Consider Net::Google CPAN Perl Module

tit Roadmap

v0.4 Beta ->v0.9 Beta

- Maintenance Releases

v1.0 (Historical Release)

- Final Release of Perl Branch

- Released approx 31 August 2009

- ▶ Once Google deprecates SOAP Search API

tit Roadmap

v2.0 (Web Application Development Branch)

■ `tcpinputtext.com`

- ▶ Google AJAX Search API
- ▶ Google Hosted Application



OWASP Testing Guide v3

4.2.1 "Spiders/Robots/Crawlers"

Christian Heinrich

christian.heinrich@owasp.org

OWASP Project Lead

SyScan'09 Singapore

Copyright © The OWASP Foundation

Permission is granted to copy, distribute and/or modify this document under the terms of the OWASP License.

The OWASP Foundation

<http://www.owasp.org>

What is a Spider/Robot/Crawler?

1. Automatically Traverses Hyperlink[s]
2. Recursively Retrieves URLs Referenced

Differentiators Between Implementations Include:

- Apply Heuristics to Selection of Hyperlink[s]
- **HTTP GET** Request at Random Time Interval

Robots Exclusion Protocol

Expected Behaviour Dictated by Web Server

- Permit/Deny Indexing, Caching, etc
 - ▶ Applies to a specific robot implementation

Specified by Two (2) Controls:

1. **<META>** tags within **HTML <HEAD>**
2. **robots.txt** within Web Root Directory

<META> Tags

<META NAME="robots" [snip]

- Applies to all robot implementations

<META NAME="googlebot" [snip]

- Applies to Googlebot

<META> Tags

[snip] content="noimageindex">

- Allows Indexing but Excludes Any Images

[snip] content="noindex">

- Prevents Indexing but Hyperlinks[s] Followed

<META> Tags

[snip] content="noindex,nofollow">

- Prevents Indexing and Following Hyperlink[s]

[snip] content="noarchive">

- Removes the "Cached" Version of the Page

robots.txt

```
cmlh$ wget http://www.google.com/robots.txt
--23:59:24-- http://www.google.com/robots.txt
=> 'robots.txt'
Resolving www.google.com... 74.125.19.103, 74.125.19.104, 74.125.19.147, ...
Connecting to www.google.com|74.125.19.103|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/plain]

[ <=> ] 3,425 --.--K/s

23:59:26 (13.67MB/s) - 'robots.txt' saved [3425]
```

User-agent: Directive

```
User-agent: *  
Allow: /searchhistory/  
Disallow: /news?output=xhtml&  
Allow: /news?output=xhtml  
Disallow: /search  
Disallow: /groups  
Disallow: /images  
...
```

"*" Applies to all Spiders/Robots/Crawlers

User-agent: Directive

```
User-agent: Googlebot
Allow: /searchhistory/
Disallow: /news?output=xhtml&
Allow: /news?output=xhtml
Disallow: /search
Disallow: /groups
Disallow: /images
...
```

"Googlebot" Applies to "Googlebot" Crawler

Disallow: Directive

```
User-agent: Googlebot
Allow: /searchhistory/
Disallow: /news?output=xhtml&
Allow: /news?output=xhtml
Disallow: /search
Disallow: /groups
Disallow: /images
...
```

Can be intentionally **ignored**:

- **Not** for Access Control within Web Server
- **Not** for Digital Rights Management (DRM)

Testing robots.txt

1. Sign into Google Webmaster Tools
2. On the Dashboard, click the URL
3. Click "*Tools*"
4. Click "*Analyze robots.txt*"

Recommendations

Implements `robots.txt` as Primary Control

- Consider `<META>` as Secondary Control
 - ▶ Supported by Minority of Spiders/Robots/Crawlers

Reference All Spiders/Robots/Crawlers

- `User-agent: *` within `robots.txt`
- `<META NAME="robots" ...`

Recommendations

Avoid Regular Expressions

- Supported by Minority of Robot Implementations
- Specify [0-9] Over 10 Individual Lines

Implement Other Technologies:

- DRM for Rights Management of Web Content
- Access Control within Web Server

Measure Continuous Improvement

- **"Analyze robots.txt"**



Search Engine Recon Continuous Improvement

Christian Heinrich

christian.heinrich@owasp.org

OWASP Project Lead

SyScan'09 Singapore

Copyright © The OWASP Foundation

Permission is granted to copy, distribute and/or modify this document
under the terms of the OWASP License.

The OWASP Foundation

<http://www.owasp.org>

Continuous Improvement

“Download Indexed Cache” is “point in time”.

- i.e. During Pen Test and/or WebAppSec Testing

Google Alerts as “new” items are indexed over time.

- Continuous Monitoring and Improvement

Near Real Time Notification

Google Alerts - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.google.com/alerts/

cmh's Bookmarks on D...

christian.heinrich@owasp.org | [Settings](#) | [FAQ](#) | [Sign out](#)

Google Alerts (BETA)

Welcome to Google Alerts

Google Alerts are email updates of the latest relevant Google results (web, news, etc.) based on your choice of query or topic.

Some handy uses of Google Alerts include:

- monitoring a developing news story
- keeping current on a competitor or industry
- getting the latest on a celebrity or event
- keeping tabs on your favorite sports teams

Create an alert with the form on the right.

You can also [click here to manage your alerts](#).

Create a Google Alert

Enter the topic you wish to monitor.

Search terms:

Type:

How often:

Deliver to:

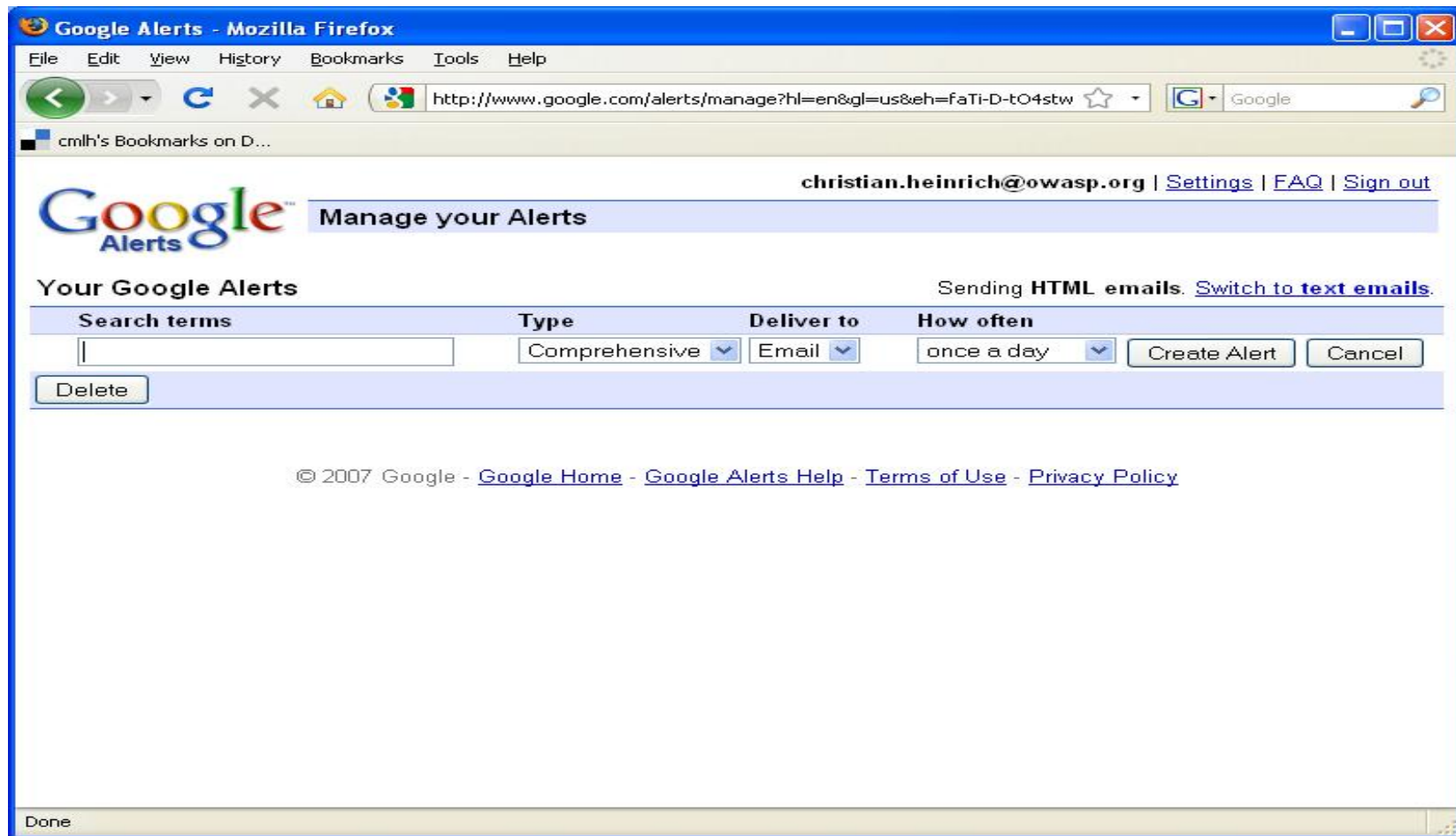
Google will not sell or share your email address.

© 2007 Google - [Google Home](#) - [Google Alerts Help](#) - [Terms of Use](#) - [Privacy Policy](#)

Done



Near Real Time Notification



Conclusion

1. Implement "Download Indexed Cache"
 - As per "Search Engine Reconnaissance/Discovery"
 - Consider Search Result "Page Ranked" Last.
2. Mitigate with "Spiders, Robots and Crawlers"
3. Implement "Google Alerts"
 - To measure Continuous Improvement
 - Continue to Repeat 1. and 2.

Closing Remarks

Upcoming Presentations:

- **Encore** OWASP Singapore Chapter July 2009 Meeting
- OWASP Melbourne, AU Chapter July 2009 Meeting

http://snipurl.com/cmlh_speaking



schedule

Closing Remarks

Thanks Thomas Lim

E-mail: `christian.heinrich@cmlh.id.au`

Slides available from:

- `http://snipurl.com/cmlh\_slideshare`